WE CLAIM

1. Apparatus for processing data, said apparatus comprising:

(i) a processor core operable to execute operations as specified by instructions of a first instruction set;

(ii) an instruction translator operable to translate instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set, at least one instruction of said second instruction set specifying an operation to be executed using one or more input variables;

(iii) an interrupt handler responsive to an interrupt signal to interrupt execution of operations corresponding to instructions of said first instruction set after completion of execution of a currently executing operation; and

(iv) restart logic for restarting execution after said interrupt; wherein

(v) said instruction translator is operable to generate a sequence of one or more sets of translator output signals corresponding to instructions of said first instruction set to represent said at least one instruction of said second instruction set, each sequence being such that no change is made to said one or more input variables until a final operation within said sequence is executed; and

(vi) after occurrence of an interrupt during execution of a sequence of operations representing said at least one instruction of said second instruction set:

(a) if said interrupt occurred prior to starting execution of a final operation in said sequence, then said restart logic restarts execution at a first operation in said sequence; and

(b) if said interrupt occurred after starting execution of a final operation in said sequence, then said restart logic restarts execution at a next instruction following said sequence.

2. Apparatus as claimed in claim 1, wherein said translator output signals include signals forming an instruction of said first instruction set.

3. Apparatus as claimed in any one of claims 1 and 2, wherein said translator output signals include control signals that control operation of said processor core and match control signals produced on decoding instructions of said first instruction set.

4. Apparatus as claimed in any one of claims 1, 2 and 3, wherein said translator output signals include control signals that control operation of said processor core and specify parameters not specified by control signals produced on decoding instructions of said first instruction set.

5. Apparatus as claimed in any one of the preceding claims, wherein said restart logic is part of said instruction translator.

6. Apparatus as claimed in any one of the preceding claims, wherein said restart logic stores a pointer to a restart location within instructions of said second instruction set that are being translated, said pointer being advanced upon execution of said final operation.

7. Apparatus as claimed in claim 6, wherein said pointer is a program counter value pointing to a memory address of a memory location storing an instruction of said second instruction set currently being translated.

8. Apparatus as claimed in any one of the preceding claims, wherein instructions of said second instruction set specify operations to be executed upon stack operands held in a stack and said input variables include input stack operands.

9. Apparatus as claimed in claim 8, wherein any stack operands removed from said stack by execution of said at least one instruction of said second instruction set are not removed until after execution of said final operation has commenced.

10. Apparatus as claimed in any one of claims 8 and 9, wherein any stack operands added to said stack by execution of said at least one instruction of said second instruction are not added until after execution of said final operation has commenced.

11. Apparatus as claimed in any one of the preceding claims, wherein said input variables include system state variables not specified within said second instruction.

12. Apparatus as claimed in any one of the preceding claims, wherein said processor has a register bank containing a plurality of registers and instructions of said first instruction set execute operations upon register operands held in said registers.

13. Apparatus as claimed in claim 12, wherein a set of registers within said register bank hold stack operands from a top potion of said stack.

14. Apparatus as claimed in claim 13, wherein said instruction translator has a plurality of mapping states in which different registers within said set of registers hold respective stack operands from different positions within said stack, said instruction translator being operable to move between mapping states when said final operation is executed so as to update said input variables.

15. Apparatus as claimed in any one of the preceding claims, wherein said instructions of said second instruction set are Java Virtual Machine instructions.

16. A method of processing data, said method comprising the steps of:

(i) executing operations as specified by instructions of a first instruction set;

(ii) translating instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set, at least one instruction of said second instruction set specifying an operation to be executed using one or more input variables;

(iii) in response to an interrupt signal, interrupting execution of operations corresponding to instructions of said first instruction set after completion of execution of a currently executing operation; and

(iv) restarting execution after said interrupt; wherein

(v) said step of translating generates a sequence of one or more sets of translator output signals corresponding to instructions of said first instruction set to represent said at least one instruction of said second instruction set, each sequence being such that no change is made to said one or more input variables until a final operation within said sequence is executed; and

(vi) after occurrence of an interrupt during execution of a sequence of operations representing said at least one instruction of said second instruction set:

(a) if said interrupt occurred prior to starting execution of a final operation in said sequence, then restarting execution at a first operation in said sequence; and

(b) if said interrupt occurred after starting execution of a final operation in said sequence, then restarting execution at a next instruction following said sequence.

17. A computer program product holding a computer program for controlling a computer to perform the method of claim 15.

18. Apparatus for data processing substantially as hereinbefore described with reference to the accompanying drawings.

19. A method of data processing substantially as hereinbefore described with reference to the accompanying drawings.

20. A computer program product holding a computer program for controlling a computer to perform a method substantially as hereinbefore described with reference to the accompanying drawings.